Retrieving Passages and Finding Answers

Mostafa Keikha, Jae Hyun Park, W. Bruce Croft, Mark Sanderson[1]
CIIR, University of Massachusetts Amherst, Amherst, MA
[1]RMIT University, Melbourne, Australia
{keikham, jhpark, croft } @cs.umass.edu, mark.sanderson@rmit.edu.au

ABSTRACT

Retrieving topically-relevant text passages in documents has been studied many times, but finding non-factoid, multiple sentence answers to web queries is a different task that is becoming increasingly important for applications such as mobile search. As the first stage of developing retrieval models for "answer passages", we describe the process of creating a test collection of questions and multiple-sentence answers based on the TREC GOV2 queries and documents. This annotation shows that most of the description-length TREC queries do in fact have passage-level answers. We then examine the effectiveness of current passage retrieval models in terms of finding passages that contain answers. We show that the existing methods are not effective for this task, and also observe that the relative performance of these methods in retrieving answers does not correspond to their performance in retrieving relevant documents.

1. INTRODUCTION

Different queries can be answered with different granularities of text. For example, "factoid" queries can be answered with single facts or named entities and navigational queries can be answered with a web page. However, "informational" queries, especially longer queries that take the form of a question, can have answers as small as a sentence or as big as multiple web pages. Our hypothesis is that there are many queries for which a passage-level, multiple-sentence answer can be superior to a document-level answer and, for those queries, result lists that include passages will be more effective than documents alone. Many of the "description" queries from TREC topics, such as "What allegations have been made about Enron's culpability in the California energy crisis?" (topic 737), are examples of the type of query for which passage-level answers should be appropriate. These queries, however, cannot be as simply categorized as the queries in factoid QA systems, nor can relatively simple templates be used to identify answers.

ADCS '14 November 27 - 28 2014, Melbourne, VIC, Australia

http://dx.doi.org/10.1145/2682862.2682877.

An area where effective passage-level answer retrieval could have broader impact is mobile search applications with limited output bandwidth based on using either a small screen or speech output. In this case, the ability to use "answer passages" to reduce the amount of output while maintaining high relevance will be critical. The popularity of recent voice search applications shows the enormous potential for an effective speech-based mobile search tool. Answer passages could also be used instead of snippets in the result page, which would help users to identify answers more quickly.

Although it is potentially an important task, answer passage retrieval has not been studied in the context of non-factoid web queries. As the first stage of a comprehensive study of this task, we build a data set based on the TREC GOV2 collection that contains passage-level answers to some of the description queries. We then explore the effectiveness of existing passage retrieval models for the task of finding answers. The main two questions we address in this paper are whether passage-level answers exist for longer web queries, and whether these passage-level answers can be found using existing passage retrieval methods.

2. RELATED WORK

Passage retrieval has been studied in the information retrieval community from different perspectives. Using passage-level information to improve document ranking has been the most common approach [3, 2]. Passages have also been used for query expansion [8] and as the first stage of factoid question answering systems [13].

The effectiveness of passage retrieval models has primarily been judged by the effectiveness of the document ranking that is generated using the passages. Less attention has been paid to directly retrieving passages as final answers to a query. This problem was partly addressed in the HARD track in TREC and the INEX ad hoc track where one of the tasks was to retrieve passages instead of documents [1, 6]. However, in these studies, the goal was to retrieve topically relevant parts of the document. A text passage that is topically relevant, however, does not necessarily provide an answer to the query. The closest effort to our study was the TREC Genomics track where the goal was to answer biomedical questions using a collection of scientific articles [5]. This study, however, was limited to the biomedical domain and there has been no similar study over open domain web queries and web collections. In this paper, we describe our effort to create a collection for answer retrieval using web documents. Further, we describe the results of current passage retrieval methods for the task of finding answers.

3. ANNOTATING ANSWER PASSAGES

Our experiments are based on the TREC GOV2 collection and its corresponding description-length queries. We hired two undergraduate students who were asked to highlight answer passages using an annotation system and a graduate student who performed quality control.

We divide topics randomly into two different groups, one for each annotator. For each topic, we retrieve the top 50 documents using the Sequential Dependence Model (SDM), a state-of-the-art retrieval model [9]. From the retrieved documents, we select the relevant documents (based on the TREC judgments) for the passage annotation phase. Each assessor annotates all the documents related to their assigned topics. They also annotated the top five documents for the other topics to study the agreement between annotators.

Guidelines were established to improve consistency. An answer passage is defined as a piece of text in a document that can answer a user information need defined in the description part of a TREC topic. Passages were evaluated based on whether they were complete (contains an entire answer) and concise (contains no irrelevant information).

Since typically there are only a few complete and concise passages with respect to each query, we relaxed the criteria somewhat and defined four levels of relevance (Perfect, Excellent, Good, Fair) based on the annotator's view of the quality of the answer passage. The definition for "excellent", as an example, was that only simple inference based on the user's background knowledge was required to answer the question using the passage, and that most of the passage is related to the answer.

Our annotators found 8,027 answer passages to 82 TREC queries, which is about 97 passages per query on average. Among all the annotated passages, 43% of them are perfect answers, 44% are excellent, 10% are good and the rest are fair answers. The annotators highlighted a total of 84,381 words in the passage answers and their term-level kappa agreement is 0.38 which is comparable to previous answer level agreements of 0.3 [11]. The average length of an answer passage is 45 words. Prior to the annotation phase, we manually filtered queries and selected the ones that are more likely to have passage-level answers. Among the 82 selected queries, only three had no annotated answers. This confirms that many web queries of this type can be answered using a small number of sentences.

4. EVALUATING ANSWER RETRIEVAL

As part of the passage retrieval task in HARD track, new evaluation measures were proposed. The proposed metrics, such as R-precision, consider relevant characters in the top retrieved passages [1]. The proposed character-level measures are generally similar to traditional document-level measures but use characters as opposed to documents. Similar character-level measures were used in the INEX ad hoc track evaluation and the TREC Genomics passage retrieval task [6, 5]. We use the Passage2 MAP measure from the TREC Genomics track for evaluating our systems. In this measure, each character is treated as a document and we calculate MAP measures based on the number of retrieved relevant characters. If a relevant character is retrieved more than once, we only consider its first retrieval as a hit. In addition to MAP, we also report character-based precision for the top 1 and 10 retrieved passages. Precision at 1 is particularly important due to the specific requirements of an answer retrieval system. In all of these evaluations, we assume that only passages marked "perfect" or "excellent" are relevant.

Table 1: Performance evaluation of QL and SDM for retrieving answers

Measure	MAP	P@1	P@10
QL	0.021	0.148	0.057
SDM	0.020	0.107	0.060
QL-Interpolated	0.022	0.073	0.062

5. PASSAGE RETRIEVAL METHODS FOR RETRIEVING ANSWERS

To study the effectiveness of existing passage retrieval methods for finding answers, we investigated different language model-based models. It has been shown previously that fixed-size window passages are the most effective for topical retrieval of documents in TREC corpora (e.g., [2]). Based on this observation and without further tuning, we fix the length of retrieved passages to 50 terms with an overlap of 25 terms. We use the two-phase passage retrieval feature in the Galago toolkit for all our experiments (www.lemur.org). Unless otherwise mentioned, we use default parameter values for existing methods such as smoothing parameters. We select the top 50 documents retrieved by the sequential dependence model (SDM) as input for our passage retrieval methods. Although we use default parameters, these values have been found to be effective in a range of previous passage and document retrieval experiments.

Query likelihood and SDM: In the first set of experiments, we employ existing language model techniques for retrieving passages. In these experiments, each passage is treated as a separate document and scored using the query likelihood model (QL) or the sequential dependence model (SDM) [9].

Further, we also experiment with an interpolation method in which we combine the score of the passage with the score of its document. We employ a homogeneity measure based on the length of the document to assign weights to each component. We use the length-based homogeneity function defined by Bendersky and Kurland that assign lower weights to document scores when document length increases [2].

Table 1 shows the performance evaluation of the language model methods. Interestingly there are no large differences between the three methods in terms of MAP and P@10. However query likelihood performs better than SDM in precision at high ranks. More interestingly, combining the document score with the passage score slightly increases the MAP and P@10 but decreases the performance at high ranks. The interpolation has a small effect mainly because our initial set of documents are all high quality documents and have very similar scores. It is worth noting that while the evaluation values are very low, they are comparable with previous studies on retrieving answers in TREC Genomics track [5].

Positional Model: In closely related work, Carmel *et. al.* used positional models (PM) for passage retrieval. They employ a $tf.idf$ scoring method for ranking passages:

$$score_{Psg}(p,q) = \sum_{t \in p \cap q} tf(t,p) \times idf(t) \qquad (1)$$

Table 2: Performance evaluation of PM for retrieving answers

Measure	MAP	P@1	P@10
PM-TFIDF	0.022	0.123	0.059
PM-Dirichlet	0.022	0.146	0.058
PM-SkewedGaussian	0.027	0.156	0.073

where tf is the query term pseudo-frequency in each passage that is estimated using positional model based on the distance between the passage and query term occurrences. The psuedo-frequency of a term t in a passage p is estimated as follows:

$$tf(t,p) = \sum_{pos \in occ(t,d)} \sum_{i=begin}^{i=end} kernel(pos, i) \qquad (2)$$

where $occ(t,d)$ is the set of all positions of t in the document, *begin* and *end* are the begin and end positions of the passage and $kernel(pos,i)$ is a kernel decay function that propagates term occurrence over all the positions in the document. In this experiment we use a Gaussian kernel function with $\sigma = 2000$, as it is shown to be an effective choice [4].

Beside the $tf.idf$ scoring model, we also use the estimated pseudo-frequencies in a query likelihood model using Dirichlet smoothing. The result for this method can be seen in table 2. As we can see, the methods have comparable performance to the other language model techniques while the query likelihood model performs slightly better than the $tf.idf$ method. Similar to our previous experiment, interpolating the document scores with the passage scores did not change our results in this experiment.

Non-symmetric Kernel Functions: In previous positional models for information retrieval, symmetric kernel functions have been employed that give the same propagation to positions before and after query term occurrences. After more in-depth investigation of the documents and queries in our collection, we found out that terms following a query term are more related to the query than terms before the query term. In order to model this property, we employ non-symmetric kernel functions that give higher value to positions after the query term occurrence. A distribution with such a kernel function is described as a distribution with positive skewness (skewness is the measure of asymmetry of a probability distribution that is zero for symmetric distributions). We employ a skew Gaussian kernel function that generalises Gaussian kernel to allow for non-zero skewness:

$$kernel_{skewed}(j,i) = e^{-\frac{(i-j)^2}{2\sigma^2}} \times \left[1 + \text{erf}(\frac{\alpha.(i-j)}{\sqrt{2}})\right] \qquad (3)$$

where *erf* is the error function and α is the skewness parameter which is set to one in our experiments. As in the previous experiment, we set the σ to 2000 and use the pseudo-frequencies in a query likelihood model. The results of this kernel function are shown in the last row of table 2. This method outperforms the others and confirms our intuition about non-symmetric kernel functions.

5.1 Query Expansion Methods in Answer Retrieval

The short length of passages makes it more likely to have a term mismatch problem between queries and answers. In other words, query terms may not occur frequently in an answer passage and other related terms will not contribute to the score. One of the possible solutions to this problem, that has been shown to be effective for document retrieval, is to expand the queries with related terms [7]. For example, the relevance model approach adds terms to the query that have high probabilities in the top retrieved documents. In this section, we study the effect of query expansion methods on the performance of answer passage retrieval. We employ relevance models (RM) to select and weight terms and we interpolate the selected terms with the original query terms, with a weight of 0.85 for original terms and 0.15 for expansion terms [7]. Without further tuning, we set the number of feedback documents and feedback terms to 25. We explore two options for selecting terms. In the first approach, we select terms from the top retrieved documents and in the second approach we select terms from the top retrieved passages. We use the expanded queries in the positional model using skewed Gaussian kernel.

Table 3: Effect of Query Expansion Methods

Measure	MAP	P@1	P@10
RM on documents	0.027	0.133	0.071
RM on passages	0.027	0.153	0.067

Table 4: Effect of input document

Measure	MAP	P@1	P@10
Top 5	0.008	0.121	0.043
Top 10	0.012	0.136	0.055
Top 25	0.023	0.169	0.074
Only Relevant	0.099	0.302	0.179

Table 3 shows the performance results of the two methods. Compared to document retrieval where query expansion can generally improve performance, neither of the two approaches outperforms their non-expanded counterparts. Even more surprisingly, document-based expansion decreases the performance, especially in the top ranks.

5.2 Quality of Initial Retrieval

In all our experiments so far, we used the top 50 retrieved documents from SDM as the input to our passage retrieval methods. In this section, we investigate if using fewer documents, presumably with higher quality, can affect the performance of our passage retrieval methods. To this end, we repeat the previous experiment with the top 5, 10 and 25 documents that are retrieved by SDM. The positional model using the skewed Gaussian kernel is the retrieval method in this experiment. The top three rows in table 4 show the results. Unsurprisingly, the MAP measure decreases when we use very few documents. However, we can see that precision at 1 and 10 is also decreased when we use 5 or 10 documents. The best result is obtained when we use the top 25 documents and precision at 1 has the most improvement. This shows that passage retrieval methods are sensitive to the quality of input documents and the top retrieved documents do not necessarily have highest quality.

Given that the document retrieval phase can have a significant effect on the passage retrieval phase, it is worth studying the upper-bound case where we have a perfect document retrieval method that returns only relevant documents. To this end, we use only our annotated documents as the input and re-run our our passage retrieval methods. The last row

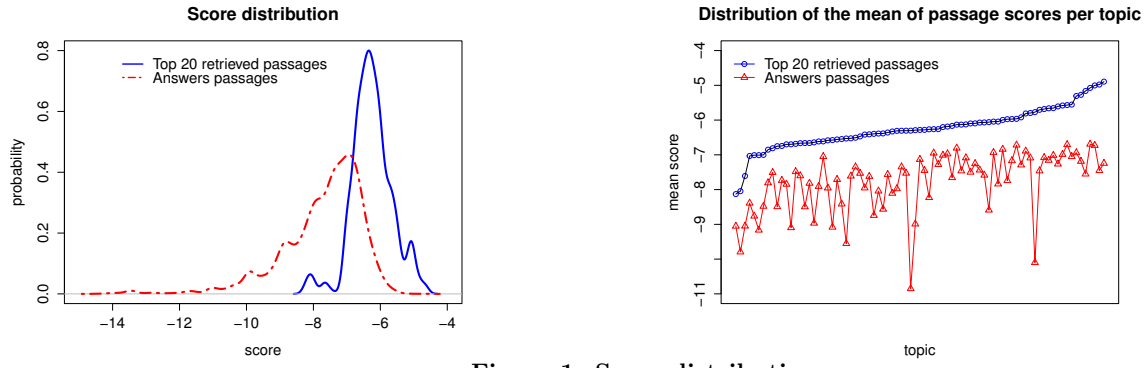

Figure 1: Score distribution

in table 4 shows the result of the experiment. As expected, all the measures are significantly improved when we retrieve passages from only relevant documents. However, while 30% of retrieved characters are relevant characters, our passage retrieval methods are still not very effective for finding the answers in the documents.

5.3 Discussion and Future Directions

Our experiments show that current passage retrieval methods that focus on topical relevance fail to perform well for answer passage retrieval. In more in-depth analysis, we looked into the scores that are assigned by our retrieval method to the annotated answers and compared them to the scores of the top 20 retrieved passages. Figure 1 shows the distribution of scores over all the topics on the left and the mean of the scores per topic on the right. Scores are the log-scale values assigned by the language model. As we can see, the answer passages have distinctively lower scores compared to the top 20 retrieved passages and this is consistent across all the topics. Interestingly there are some answers, mostly with scores less than -11, which do not contain any of the query terms.

The results show that features based on term frequencies are not sufficient for retrieving answer passages. Although the results could be improved somewhat by more intensive tuning, we believe that there is an obvious need to incorporate other types of features into retrieval models. We are currently exploring a range of features, including linguistic features studied for CQA data [12] and features used in generating summaries [10].

6. ACKNOWLEDGMENTS

This work was supported in part by IBM subcontract 4913003298 under DARPA prime contract HR001-12-C-0015, and by the Australian Research Council (DP140102655). Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

7. REFERENCES

[1] J. Allan. HARD track overview in TREC 2004 - high accuracy retrieval from documents. In *Proceedings of TREC*, 2004.

[2] M. Bendersky and O. Kurland. Utilizing passage-based language models for document retrieval. In *Proceedings of ECIR'08*, pages 162–174, 2008.

[3] J. Callan. Passage-level evidence in document retrieval. In *Proceedings of SIGIR'94*, pages 302–310, 1994.

[4] D. Carmel, A. Shtok, and O. Kurland. Position-based contextualization for passage retrieval. In *Proceedings of CIKM'13*, pages 1241–1244. ACM, 2013.

[5] W. R. Hersh, A. M. Cohen, P. M. Roberts, and H. K. Rekapalli. TREC 2006 genomics track overview. In *Proceedings of TREC'06*, 2006.

[6] J. Kamps, J. Pehcevski, G. Kazai, M. Lalmas, and S. Robertson. INEX 2007 evaluation measures. In *proceedings of INEX workshop*, pages 24–33, 2007.

[7] V. Lavrenko and W. B. Croft. Relevance based language models. In *Proceedings of ACM SIGIR'01*, pages 120–127. ACM, 2001.

[8] X. Liu and W. Croft. Passage retrieval based on language models. In *Proceedings of CIKM*, pages 375–382, 2002.

[9] D. Metzler and W. Croft. A Markov random field model for term dependencies. In *Proceedings of SIGIR*, pages 472–479, 2005.

[10] D. Metzler and T. Kanungo. Machine learned sentence selection strategies for query-biased summarization. In *proceedings of SIGIR workshop on Learning to Rank for Information Retrieval*. ACM, 2008.

[11] T. Sakai, D. Ishikawa, N. Kando, Y. Seki, K. Kuriyama, and C.-Y. Lin. Using graded-relevance metrics for evaluating community QA answer selection. In *Proceedings of WSDM'11*, pages 187–196, 2011.

[12] M. Surdeanu, M. Ciaramita, and H. Zaragoza. Learning to rank answers to non-factoid questions from web collections. *Computational Linguistics*, 37(2):351–383, 2011.

[13] D. Zhang and W. Lee. A language modeling approach to passage question answering. In *Proceedings of TREC*, pages 489–495, 2003.